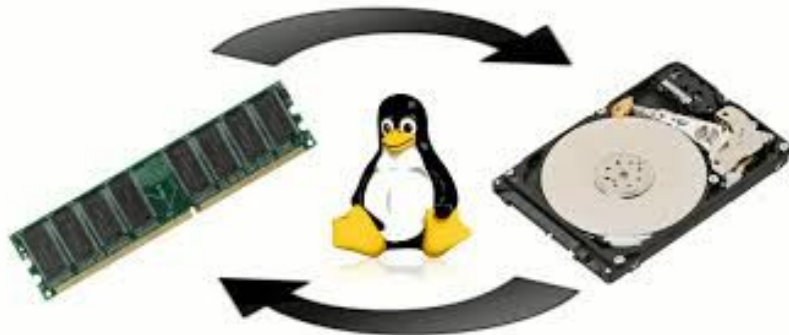


# Unidad 3

## GESTIÓN DE MEMORIA



**Cátedra:** ARQUITECTURA Y SISTEMAS OPERATIVOS

**Docente:** Ing. Guillermo Andrés Martínez

**e-mail:** gamartinez@udc.edu.ar

**Año:** 2025

# Memoria

**DEFINICIÓN:** dispositivo que retiene, memoriza o almacena datos informáticos durante algún de tiempo.



# INTRODUCCIÓN

Se pueden definir algunos parámetros generales aplicables a todas las memorias

- **Unidad de almacenamiento:** Bit.
- **Capacidad de almacenamiento:** Cantidad de bits que puede almacenarse. Si bien la unidad de almacenamiento es el **bit**, muchas veces se usa el **byte**.

## Unidades de Medidas de Almacenamiento

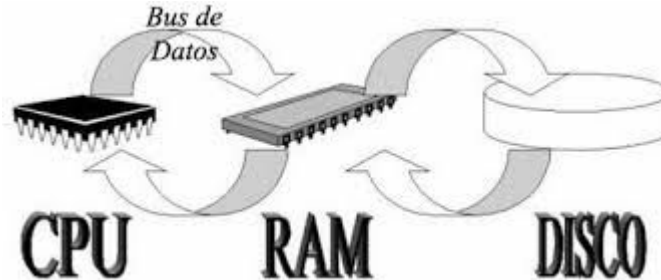
Medida	Simbologia	Equivalencia	Equivalente en Bytes
byte	b	8 bits	1 byte
kilobyte	Kb	1024 bytes	1 024 bytes
megabyte	MB	1024 KB	1 048 576 bytes
gigabyte	GB	1024 MB	1 073 741 824 bytes
terabyte	TB	1024 GB	1 099 511 627 776 bytes
Petabyte	PB	1024 TB	1 125 899 906 842 624 bytes
Exabyte	EB	1024 PB	1 152 921 504 606 846 976 bytes
Zetabyte	ZB	1024 EB	1 180 591 620 717 411 303 424 bytes
Yottabyte	YB	1024 ZB	1 208 925 819 614 629 174 706 176 bytes
Brontobyte	BB	1024 YB	1 237 940 039 285 380 274 899 124 224 bytes
Geopbyte	GB	1024 BB	1 267 650 600 228 229 401 496 703 205 376 bytes

# INTRODUCCIÓN

- Las memorias son los dispositivos que almacenan datos e instrucciones en una computadora.
- Desafortunadamente, memorias de velocidades similares al procesador son muy caras. Por esta razón la información almacenada se distribuye en forma compleja en una variedad de memorias diferentes, con características físicas distintas.

Una clasificación funcional de las memorias es la siguiente:

- **Memoria interna:** Constituida por los registros internos de la CPU. Tiene alta velocidad.
- **Memoria central (o principal o RAM):** Almacena programas y datos, es relativamente grande, rápida y es accedida directamente por la CPU a través de un bus.
- **Memoria secundaria:** Se usa para el almacenamiento de programas del sistema y grandes archivos. Su capacidad es mucho mayor que las anteriores, pero más lenta y el acceso a la misma por parte de la CPU es indirecto. Las principales tecnologías son la magnética y la óptica.



# INTRODUCCIÓN

**Tiempo de acceso ( $t_a$ ):** Es el que se tarda en leer o escribir una palabra en la memoria desde el momento que se direcciona. La velocidad de acceso  $b_a = 1/t_a$  se mide en palabras/segundo.

**Tipo de acceso:**

- **Acceso aleatorio:** cuando el tiempo de acceso es similar para cualquier posición
- **Acceso serie:** cuando el tiempo de acceso depende de la posición que ocupa la palabra dentro de la memoria.

**Tiempo de ciclo ( $t_c$ ):** Indica el mínimo tiempo entre dos accesos sucesivos a la memoria. El tiempo  $t_c$  es mayor que el tiempo  $t_a$ .

**Medio físico**

- **Electrónicas:** construidas con semiconductores.
- **Magnéticas:** basadas en el fenómeno de histéresis de los materiales ferromagnéticos.
- **Ópticas:** utilizan la tecnología láser.



# INTRODUCCIÓN

## **Estabilidad**

- **Volatilidad:** el contenido de la memoria se pierde cuando se suspende la alimentación eléctrica.
- **Almacenamiento dinámico:** El bit se almacena como carga de una capacidad parásita de un transistor MOS. La información se pierde cuando el capacitor se descarga lo que hace necesario un refresco periódico para restaurar el contenido antes que se deteriore.
- **Lectura destructiva (DRO):** Al efectuar la lectura se pierde la información, por lo cual dicho proceso debe acompañarse de una restauración.

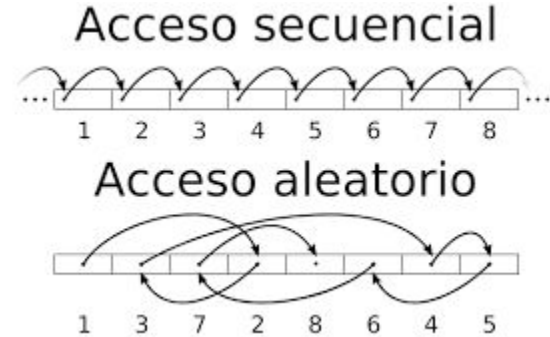


# Características de las memorias

## Accesibilidad secuencial o aleatoria a información

Dependiendo de la habilidad para acceder a información contigua o no, se puede clasificar en:

- **Acceso aleatorio:** significa que se puede acceder a cualquier localización de la memoria en cualquier momento en el mismo intervalo de tiempo, normalmente pequeño.
- **Acceso secuencial:** significa que acceder a una unidad de información tomará un intervalo de tiempo variable, dependiendo de la unidad de información que fue leída anteriormente. El dispositivo puede necesitar buscar (posicionar correctamente el cabezal de lectura/escritura de un disco), o dar vueltas (esperando a que la posición adecuada aparezca debajo del cabezal de lectura/escritura en un medio que gira continuamente).



# Características de las memorias

## Habilidad para cambiar la información

- Las memorias de lectura/escritura o memorias cambiables permiten que la información se reescriba en cualquier momento.
- La memoria de sólo lectura (Read-Only Memory, ROM) retiene la información almacenada en el momento de fabricarse. Un ejemplo son los CD-ROM.
- Las memorias de escritura lenta y lectura rápida son memorias de lectura/escritura que permite que la información se reescriba múltiples veces pero con una velocidad de escritura mucho menor que la de lectura. Un ejemplo son los CD-RW.





# Características de las memorias

## Direccionamiento de la información

- En la memoria de localización direccionable, cada unidad de información accesible individualmente en la memoria se selecciona con su dirección de memoria numérica. Ejemplo Memoria RAM.
- En las memorias de sistema de archivos, la información se divide en archivos informáticos de longitud variable y un fichero concreto se localiza en directorios y nombres de archivos «legible por humanos».
- En las memorias de contenido direccionable (content-addressable memory), cada unidad de información legible individualmente se selecciona con un valor hash o un identificador corto sin relación con la dirección de memoria en la que se almacena la información. La memoria de contenido direccionable pueden construirse usando software o hardware; la opción hardware es la opción más rápida y cara.

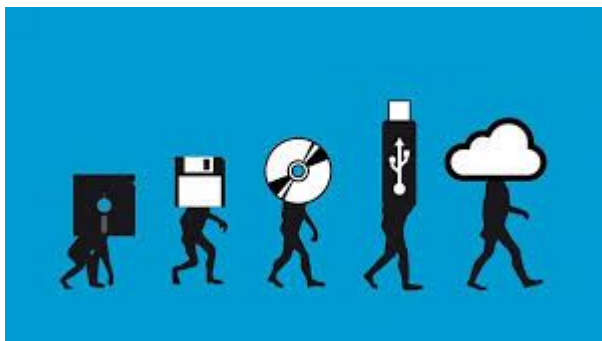


# Características de las memorias

## Capacidad de memoria

Memorias de mayor capacidad son el resultado de la rápida evolución en tecnología de materiales semiconductores.

La velocidad de los computadores se incrementó, multiplicada por 100.000 aproximadamente y la capacidad de memoria creció en una proporción similar.



# Tecnologías, dispositivos y medios

## **Memoria de semiconductor**

La memoria de semiconductor usa circuitos integrados basados en semiconductores para almacenar información. Un chip de memoria de semiconductor puede contener millones de minúsculos transistores o condensadores. Existen memorias de semiconductor de ambos tipos: volátiles y no volátiles. En las computadoras modernas, la memoria principal consiste casi exclusivamente en memoria de semiconductor volátil y dinámica, también conocida como memoria dinámica de acceso aleatorio o más comúnmente RAM.



# Tecnologías, dispositivos y medios

## Memoria magnética

Las memorias magnéticas usan diferentes patrones de magnetización sobre una superficie cubierta con una capa magnetizada para almacenar información. Las memorias magnéticas son no volátiles. Se llega a la información usando uno o más cabezales de lectura/escritura. Como el cabezal de lectura/escritura solo cubre una parte de la superficie, el almacenamiento magnético es de acceso secuencial y debe buscar, dar vueltas o las dos cosas. En 'computadoras modernas', la superficie magnética es de alguno de estos tipos:

- Disco magnético.
  - Disquete, usado para memoria fuera de línea.
  - Disco duro, usado para memoria secundario.
- Cinta magnética, usada para memoria terciaria y fuera de línea.



# Tecnologías, dispositivos y medios

## Memoria de disco óptico

Las memorias en disco óptico almacenan información usando agujeros minúsculos grabados con un láser en la superficie de un disco circular. La información se lee iluminando la superficie con un diodo láser y observando la reflexión. Los discos ópticos son no volátil y de acceso secuencial. Los siguientes formatos son de uso común:

- **CD, CD-ROM, DVD:** Memorias de simplemente solo lectura, usada para distribución masiva de información digital (música, vídeo, programas informáticos).
- **CD-R, DVD-R, DVD+R:** Memorias de escritura única usada como memoria terciaria y fuera de línea.
- **CD-RW, DVD-RW, DVD+RW, DVD-RAM:** Memoria de escritura lenta y lectura rápida usada como memoria terciaria y fuera de línea.
- **Blu-ray:** Formato de disco óptico pensado para almacenar vídeo de alta calidad y datos. Para su desarrollo se creó la BDA, en la que se encuentran, entre otros, Sony o Phillips.



# Jerarquía de almacenamiento

Una PC contiene varios tipos de almacenamiento, cada uno con su propósito individual.



# Almacenamiento Primario

Contiene los programas en ejecución y los datos con que operan. Se puede transferir información muy rápidamente entre un registro del microprocesador y localizaciones del almacenamiento principal. Está directamente conectada a la CPU a través de buses de direcciones, datos y control. Todo dato contenido en memoria debe poder encontrarse basándose en su dirección. La información es Volátil.

DDR



DDR2



DDR3



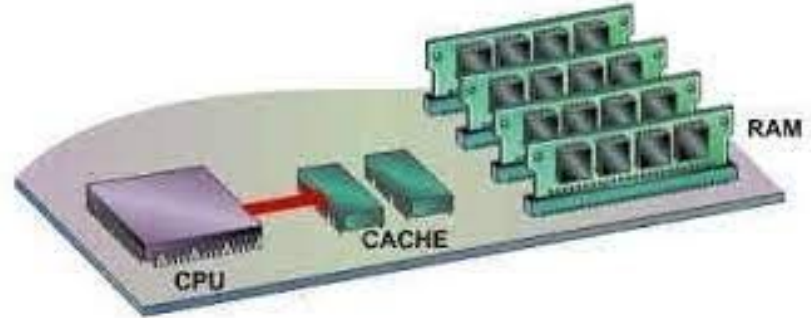
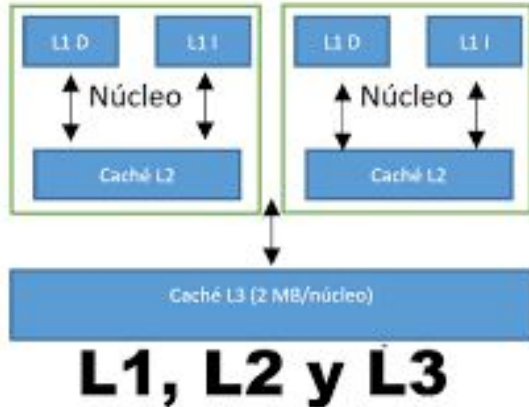
DDR4



# Almacenamiento Primario- . Cache

La gran diferencia de velocidad entre el procesador y la memoria primaria dio origen a la memoria caché. Esta es una memoria de muy alta velocidad, típicamente entre 10 y 100 veces más que la memoria primaria, y se emplea para mejorar la eficiencia o rendimiento del CPU. Parte de la información de la memoria principal se duplica en la memoria caché. Comparada con los registros, la caché es ligeramente más lenta, pero de mayor capacidad. Sin embargo, es más rápida, aunque de mucha menor capacidad que la memoria principal.

## Memoria caché





# Almacenamiento Secundario

Tiene mayor capacidad que la memoria primaria, pero es mucho más lenta.  
Se usa para guardar información en forma persistente.



disco duro



DVD



CD



pendrive



tarjeta SD



Memory Stick



disco duro portátil



Disquete

# Almacenamiento Terciario

La memoria terciaria es un medio de almacenamiento masivo fuera de línea según lo solicite el sistema operativo de la computadora. La memoria terciaria se usa par realizar backups.



# Almacenamiento fuera de línea

El almacenamiento fuera de línea (off-line) es un sistema donde el medio de almacenamiento puede ser extraído fácilmente del dispositivo de almacenamiento. Estos medios de almacenamiento suelen usarse para transporte y archivo de datos. En computadoras modernas son de uso habitual para este propósito los disquetes, discos ópticos y las memorias flash, incluyendo las unidades USB. También hay discos duros USB que se pueden conectar rápidamente. Los dispositivos de almacenamiento fuera de línea usados en el pasado son cintas magnéticas en muchos tamaños y formatos diferentes, y las baterías extraíbles de discos Winchester.



# Almacenamiento de red

El almacenamiento de red es cualquier tipo de almacenamiento de computadora que incluye el hecho de acceder a la información a través de una red informática. El almacenamiento en red incluye:

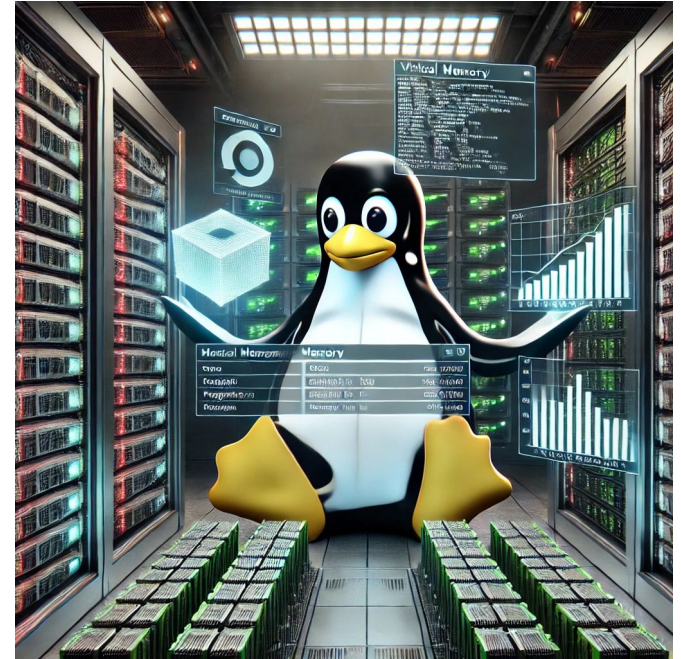
- El almacenamiento asociado a red es una memoria secundaria o terciaria que reside en una computadora a la que otra de éstas puede acceder a través de una red de área local, una red de área extensa, una red privada virtual o, en el caso de almacenamiento de archivos en línea, internet.
- Las redes de computadoras son computadoras que no contienen dispositivos de almacenamiento secundario. En su lugar, los documentos y otros datos son almacenados en un dispositivo de la red.



# Administración de Memoria

La parte del S. O. que administra la memoria se llama “administrador de la memoria”:

- Lleva un registro de las partes de memoria que se están utilizando y de aquellas que no.
- Asigna espacio en memoria a los procesos cuando estos la necesitan.
- Libera espacio de memoria asignada a procesos que han terminado.



# Memoria Virtual

Es una técnica de gestión de memoria que permite a un sistema operativo utilizar más memoria de la que físicamente tiene disponible, usando el almacenamiento secundario o Hard Disk.

Facilita la ejecución de programas grandes y mejora el aislamiento y protección de procesos.



# Técnicas de Memoria Virtual

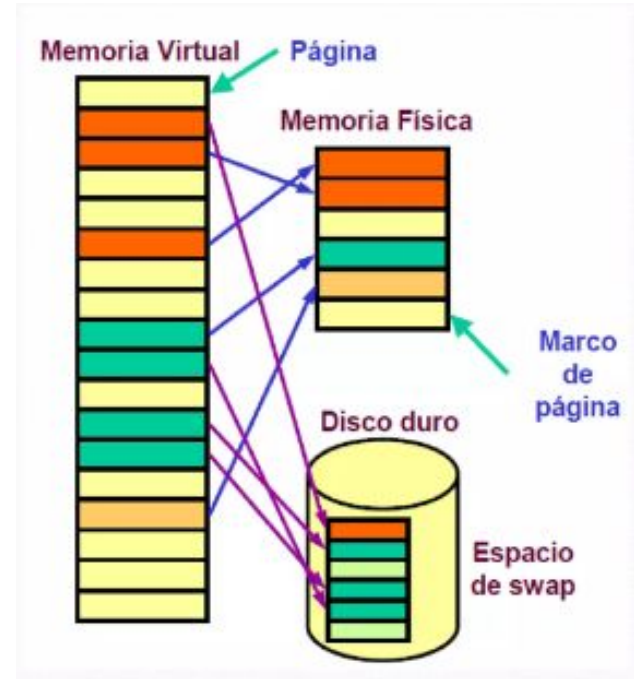
Los métodos más comunes de implementación son mediante:

- Técnicas de “paginación”.
- Técnicas de “segmentación”.
- Una combinación de ambas técnicas (Segmentación Paginada).



# Paginación

Es una técnica que divide la memoria en bloques pequeños llamados **páginas**. Estas páginas se mueven entre el almacenamiento secundario (disco) y el primario (RAM) en marcos de página de igual tamaño. Permite que los programas usen más memoria de la que físicamente está disponible, al cargar solo las páginas necesarias, mejorando la eficiencia en el uso de la memoria.

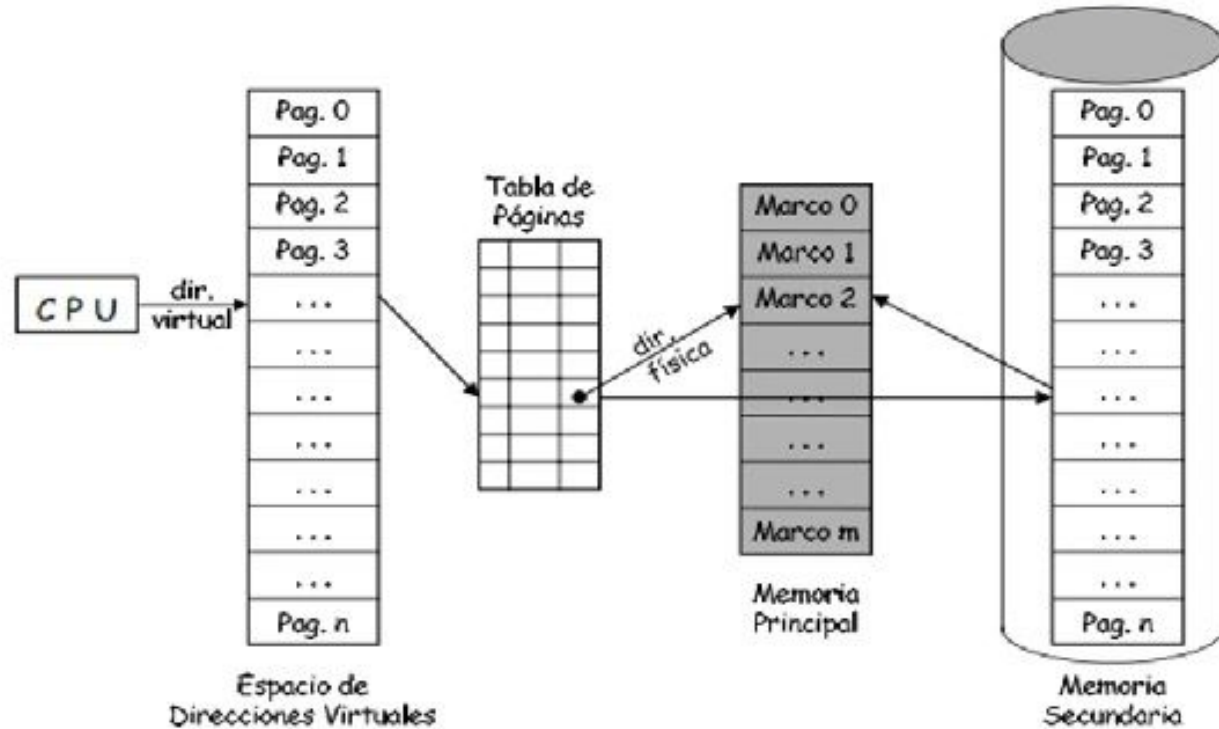




# Paginación

Memoria Virtual

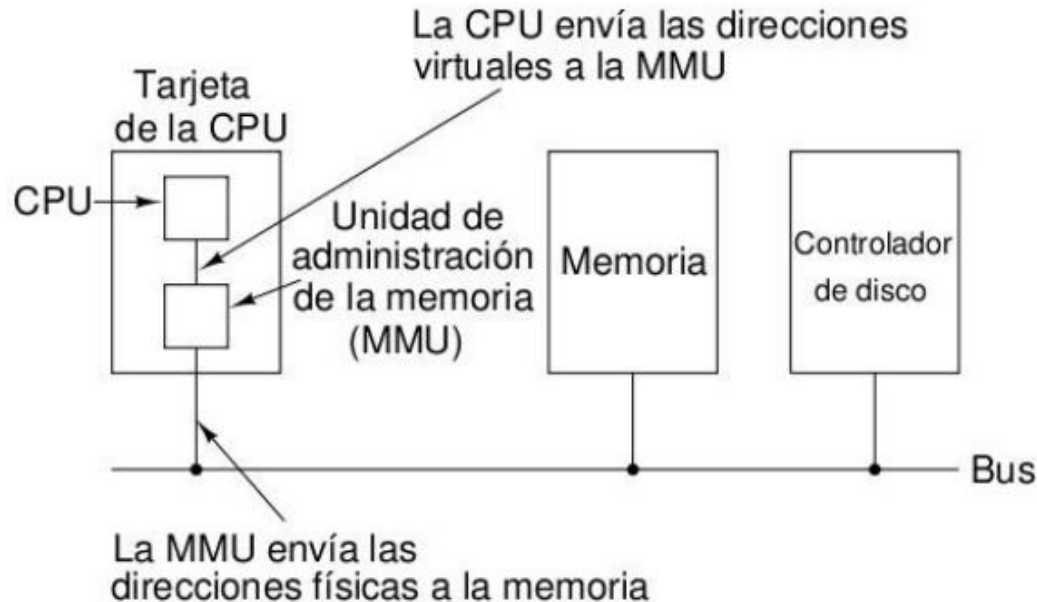
Paginación



# Paginación

Cuando un programa ejecuta una instrucción como `MOVE REG,1000` está copiando el contenido de la dirección de memoria 1000 en REG.

1000 es una dirección virtual, esta no pasan directamente al bus de memoria; en vez de ello, se envían a una unidad de administración de memoria (MMU), un chip que transforma las direcciones virtuales en direcciones de memoria física como se ilustra en la siguiente figura.



# Transformación de páginas

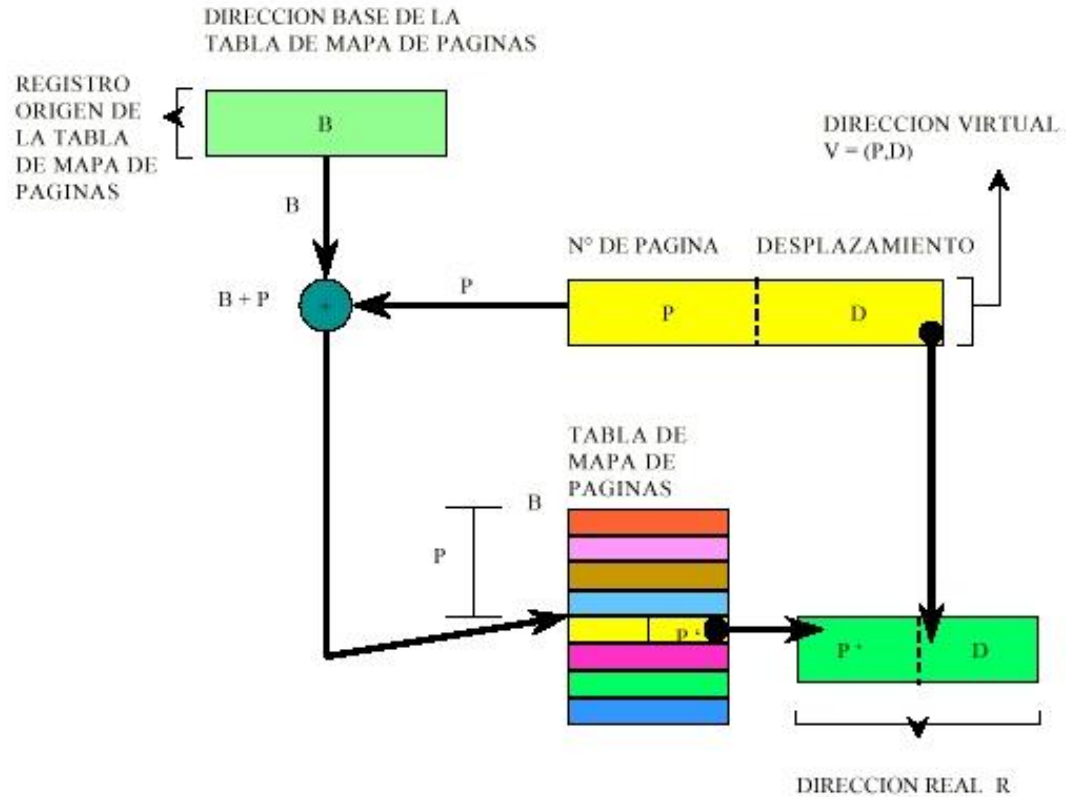
Las direcciones virtuales se representan como un par "(p, d)", donde **p** es el número de página y **d** el desplazamiento dentro de la página. Para traducir la dirección virtual a una dirección física:

1. El sistema usa una "tabla de mapa de páginas" para cada proceso, almacenada en memoria física.
2. El procesador accede a esta tabla mediante un "registro origen de la tabla de páginas" (B).
3. Se suma el número de página "P" a la dirección base de la tabla para localizar la dirección física de la Página".
4. Luego, se suma el desplazamiento "D" a "P' " para obtener la dirección física final: " $R = P' + D$ ".

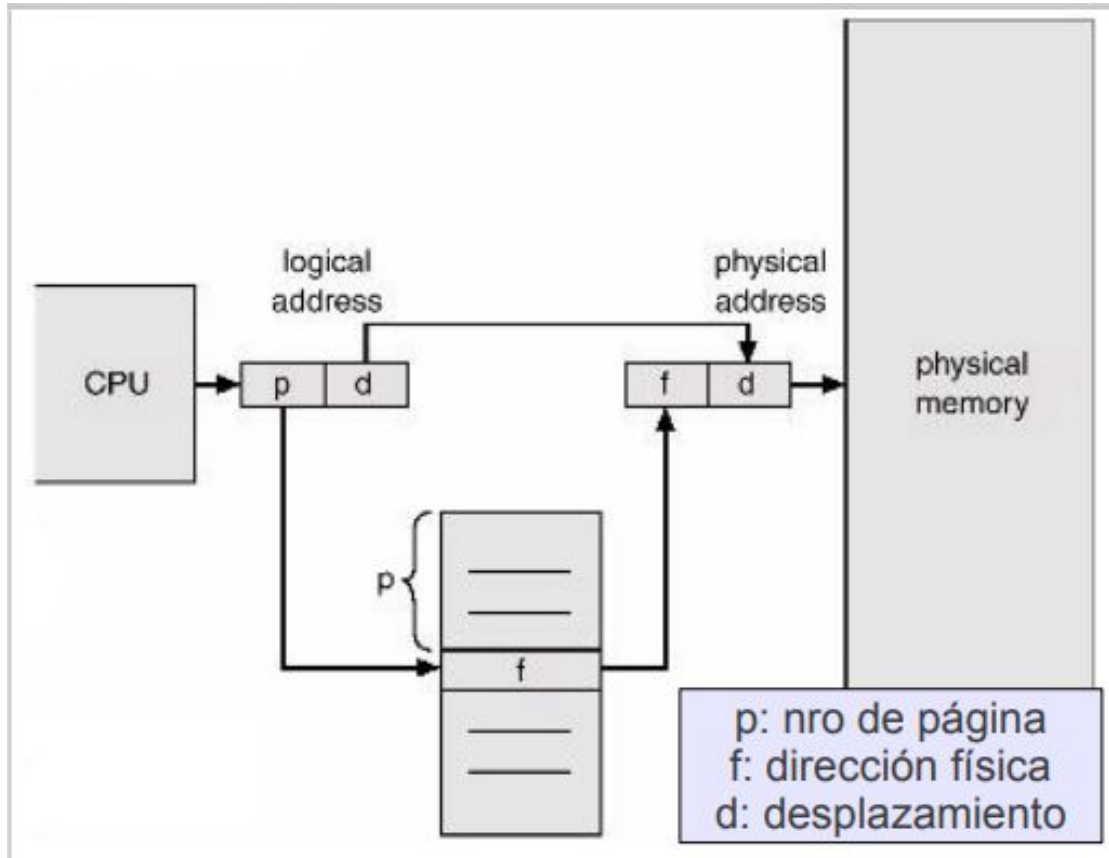
Esto traduce cada dirección virtual en su equivalente física en memoria.



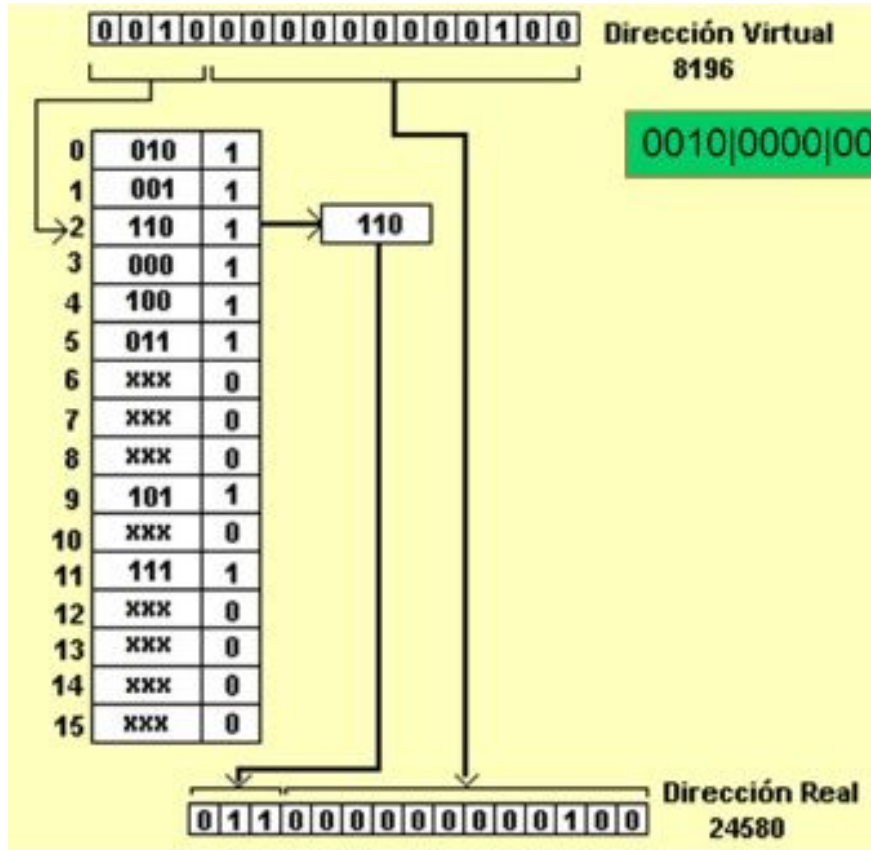
# Transformación de bloques en forma directa



# Tabla de Página

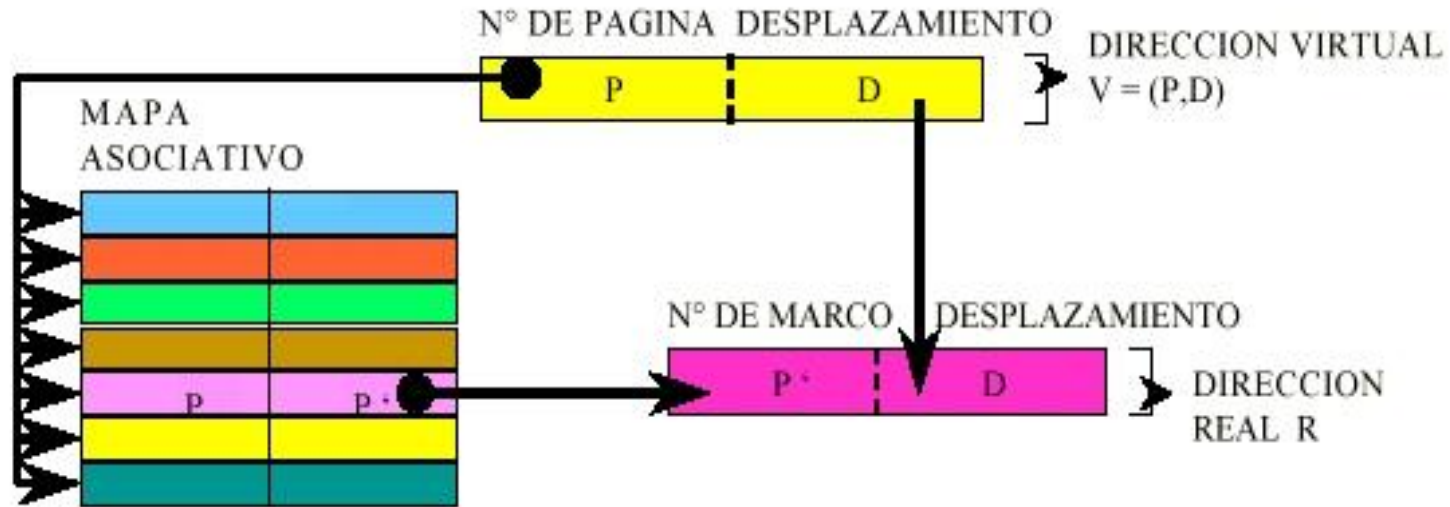


# Tabla de Página

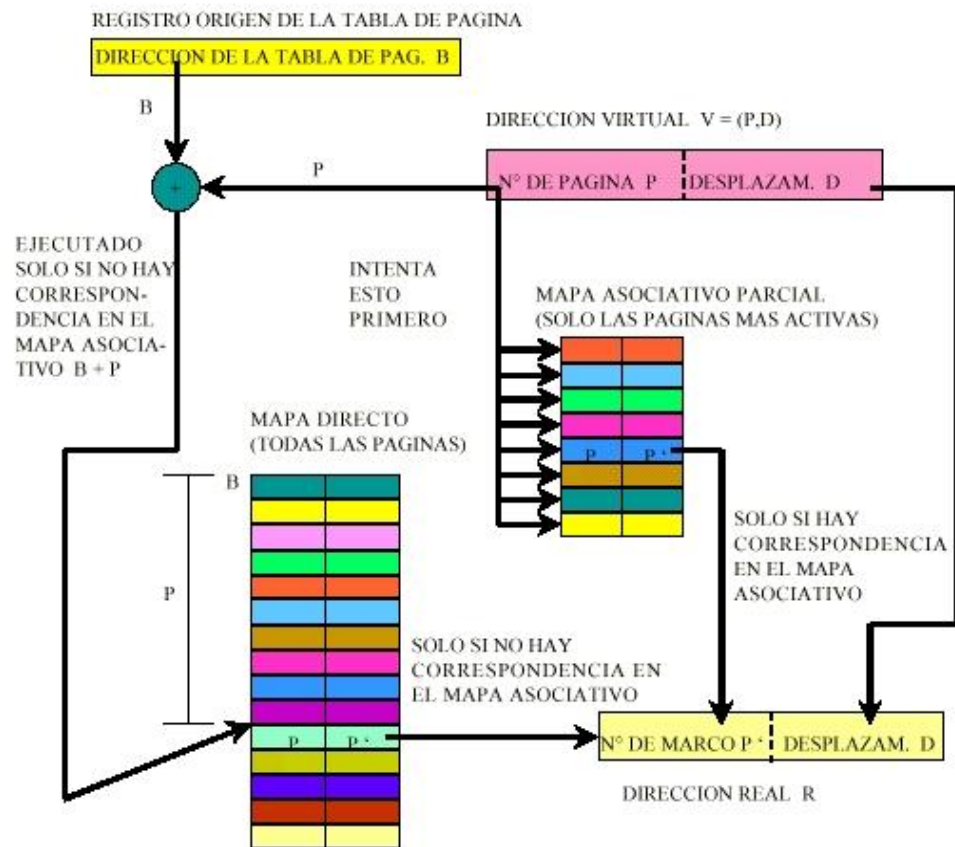


El número de página virtual sirve como índice para consultar la tabla de páginas y encontrar la entrada correspondiente a esa página virtual. El propósito de la tabla de páginas es transformar páginas virtuales en marcos de página.

# Transformación de bloques en forma asociativa



# Traducción de Direcciones de Paginación Transformación Asociativa / Directa



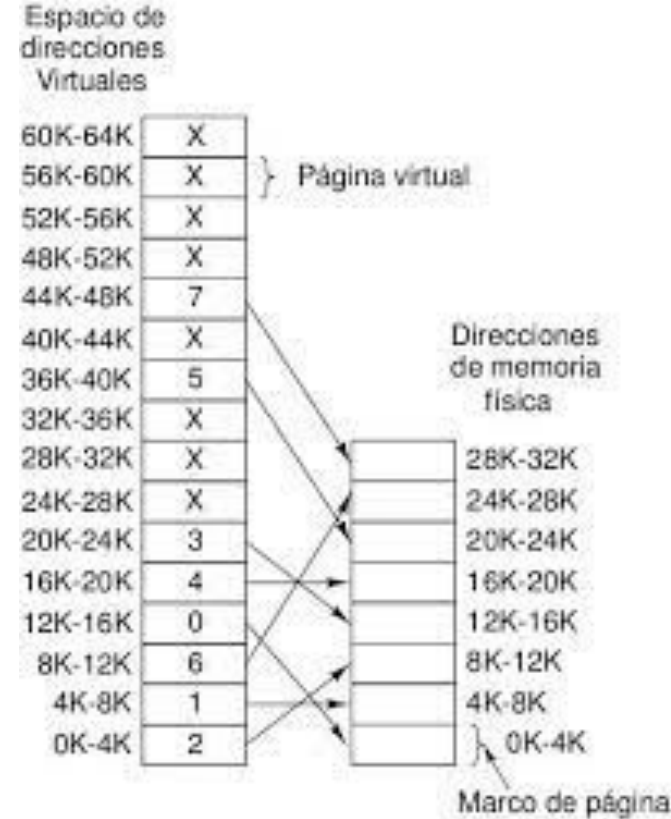


# Ejemplo de Paginación

En este ejemplo, las páginas son de 4k.

Cuando el proceso trata de acceder a la dirección 0, mediante `MOVE REG,0` la dirección virtual 0 se envía a la MMU. La MMU ve que esta dirección virtual queda en la página 0 (0 a 4095) que, según su transformación, es el marco de página 2 (8192 a 12287). La MMU transforma la dirección a 8192 y la coloca en el bus.

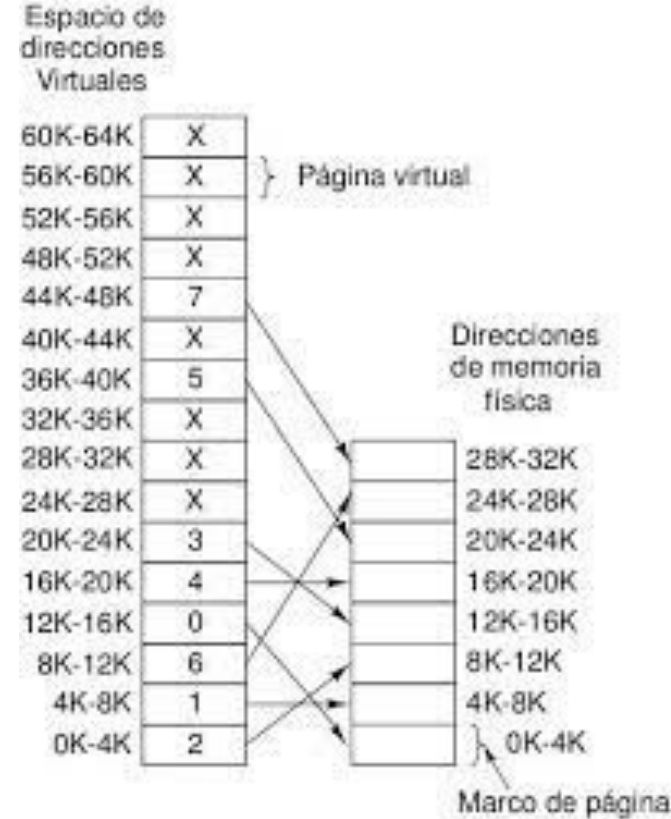
La MMU ha transformado efectivamente todas las direcciones virtuales entre 0 y 4095 en las direcciones físicas 8192 a 12287.



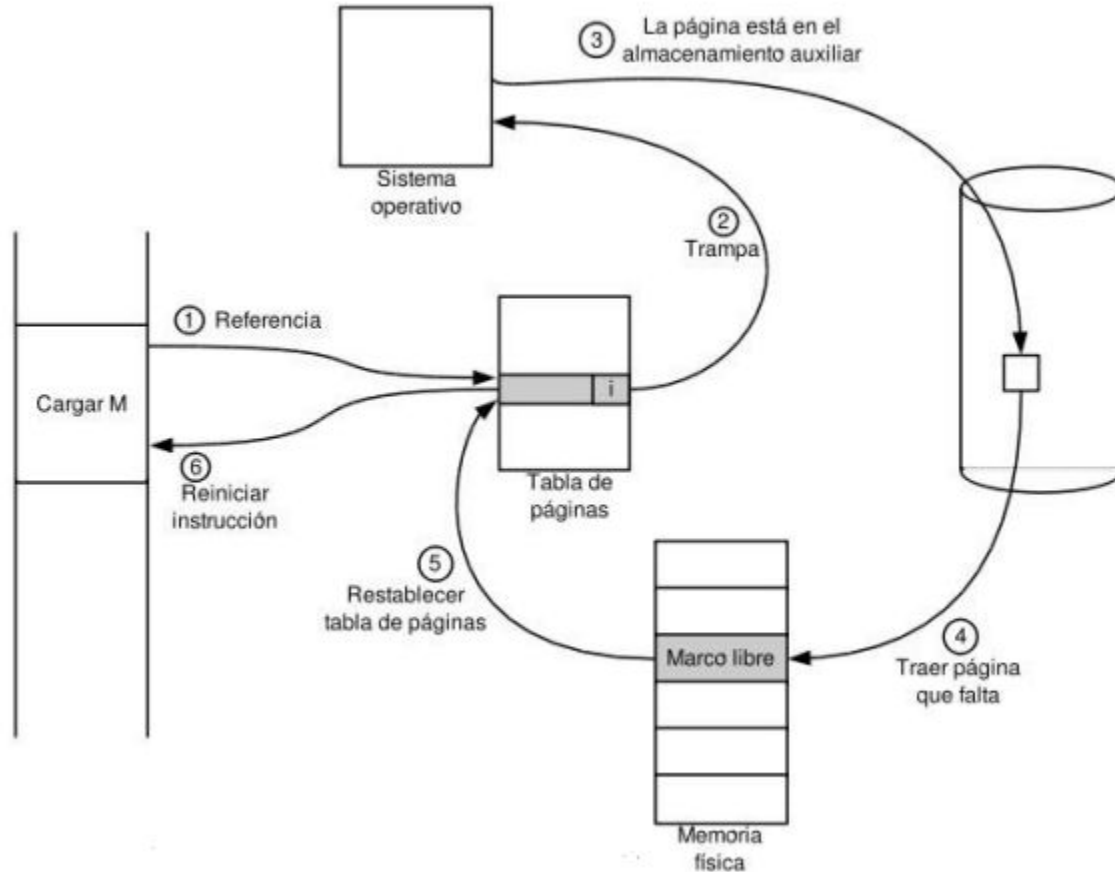
# Falla de Página

¿Qué sucede si el programa trata de usar una página sin correspondencia?

Ejemplo: `MOVE REG,32780` La MMU se da cuenta de que la página no tiene correspondencia (X) y hace que la CPU pase el control por una trampa al sistema operativo, llamado falla de página. El SO escoge un marco de página poco utilizado y escribe su contenido de vuelta en el disco; luego trae la página a la que se hizo referencia y la coloca en el marco de página recién liberado, modifica el mapa, y reinicia la instrucción atrapada.

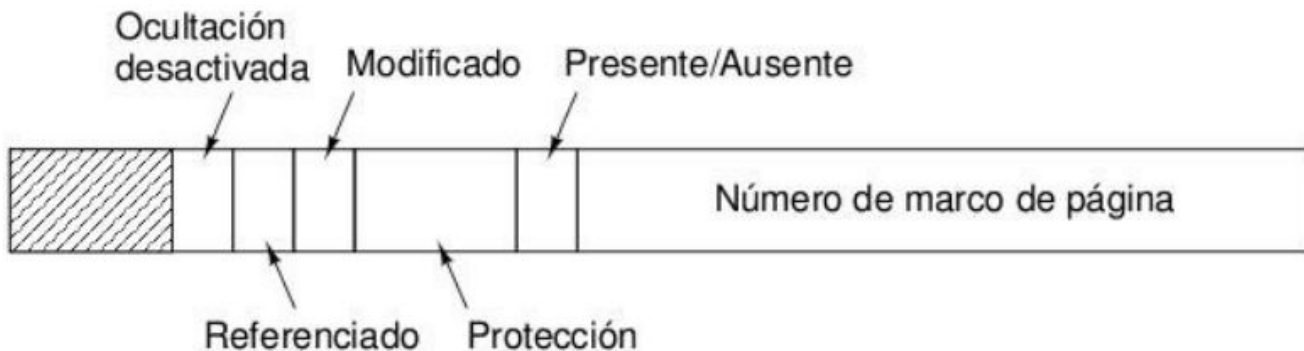


# Falla de Página



# Entrada Tabla de Página

- **Número de marco de página**
- **Bit Presente/ausente:** Si este bit es 1, la entrada es válida, si es 0, la página virtual a la que la entrada pertenece no está actualmente en la memoria. Se produce una falla de página.
- **Los bits de Protección:** 0 significa lectura/escritura mientras que 1 significa sólo lectura.
- **Los bits Modificada y Referida:** 1 si fue modificada y referida y 0 Si no modificada o referida.
- **El bit Referida:** se enciende cada vez que se hace referencia a una página, sea para lectura o escritura.
- **Ocultismo o Cache inhabilitada:** con este bit, puede desactivarse la colocación en caché.



# ALGORITMOS DE SUSTITUCIÓN DE PÁGINAS

Cuando ocurre una falla de página, el sistema operativo tiene que escoger la página que sacará de la memoria para que pueda entrar la nueva página.

Las principales algoritmos son:

1. **Reposición de páginas al azar.**
2. **FIFO - Primero en entrar - primero en salir.**
3. **Segunda oportunidad**
4. **NRU - No usada Recientemente.**



# ALGORITMO REPOSICIÓN AL AZAR

Consiste en escoger al azar la página que va a ser reemplazada.

Todas las páginas del almacenamiento principal deben tener la misma probabilidad de ser reemplazadas.

Este esquema es raramente usado.



# ALGORITMO FIFO

El nuevo se coloca al final de la lista; el que está a la cabeza de la lista se elimina.

El S.O. mantiene una lista de todas las páginas que están en la memoria, siendo la página que está a la cabeza de la lista la más vieja, y la del final, la más reciente.

Cuando hay una falla de página, se elimina la página que está a la cabeza de la lista y se agrega la nueva página al final.



# ALGORITMO DE SEGUNDA OPORTUNIDAD

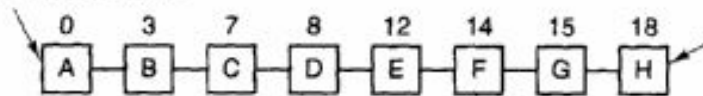
Evita el problema de desalojar una página muy utilizada.

Consiste en inspeccionar el bit R de la página más vieja.

Si es 0, sabremos que la página, además de ser vieja, no ha sido utilizada recientemente, así que la reemplazamos de inmediato.

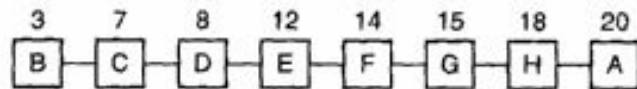
Si el bit R es 1, se apaga el bit, se coloca la página al final de la lista de páginas, y se actualiza su tiempo de carga como si acabara de ser traída a la memoria. Luego continúa la búsqueda.

Página que se cargó primero



Página que se cargó más recientemente

(a)



A se trata como página recién cargada

(b)



# ALGORITMO NRU PÁGINA NO USADA RECIENTEMENTE

El SO examina las páginas y las divide en cuatro categorías con base en base a los bits R y M:

- **Clase 0:** no referida, no modificada. 00
- **Clase 1:** no referida, modificada. 01
- **Clase 2:** referida, no modificada. 10
- **Clase 3:** referida, modificada. 11

El algoritmo NRU elimina la página que tiene el número más bajo.



# ALGORITMO SUSTITUCIÓN DE PÁGINAS POR RELOJ

El algoritmo de segunda oportunidad es innecesariamente eficiente porque constantemente cambia de lugar páginas dentro de su lista. Un enfoque mejor consiste en mantener todas las páginas en una lista circular con forma de reloj. Una manecilla apunta a la página más vieja. |

Cuando ocurre una falla de página, se inspecciona la página a la que apunta la manecilla. Si su bit R es 0, se desaloja la página, se inserta la nueva página en el reloj en su lugar, y la manecilla avanza una posición. Si R es 1, se pone en 0 y la manecilla se avanza a la siguiente página. Este proceso se repite hasta encontrar una página con  $R = 0$ . No resulta sorprendente que este algoritmo se llame de reloj. La única diferencia respecto al de segunda oportunidad es la implementación.

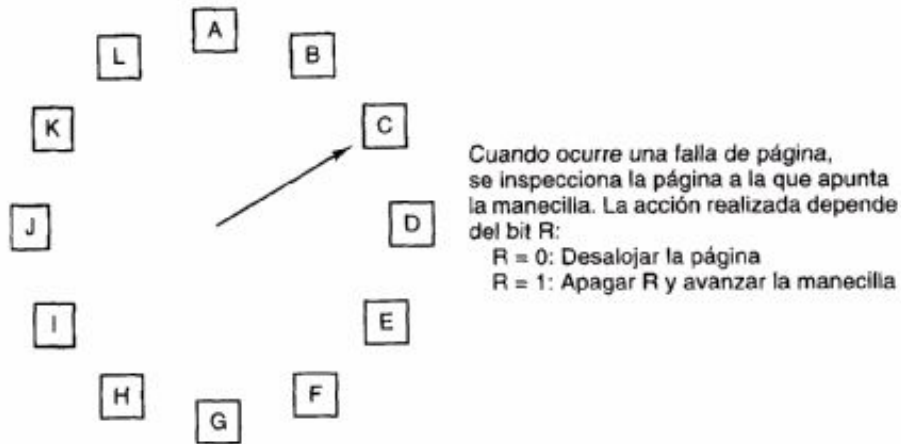
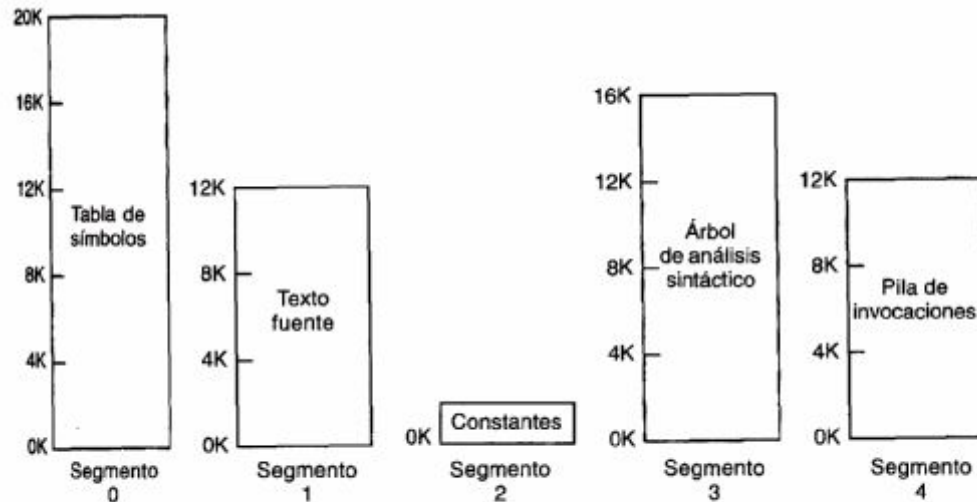


Figura 4-14. Algoritmo de reemplazo de páginas por reloj.

# Segmentación

Es una técnica de gestión de memoria donde la memoria se divide en segmentos lógicos (como código, datos, pila). Cada segmento puede tener un tamaño diferente y se almacena en partes no contiguas de la memoria. Es más, la longitud de los segmentos puede cambiar durante la ejecución.

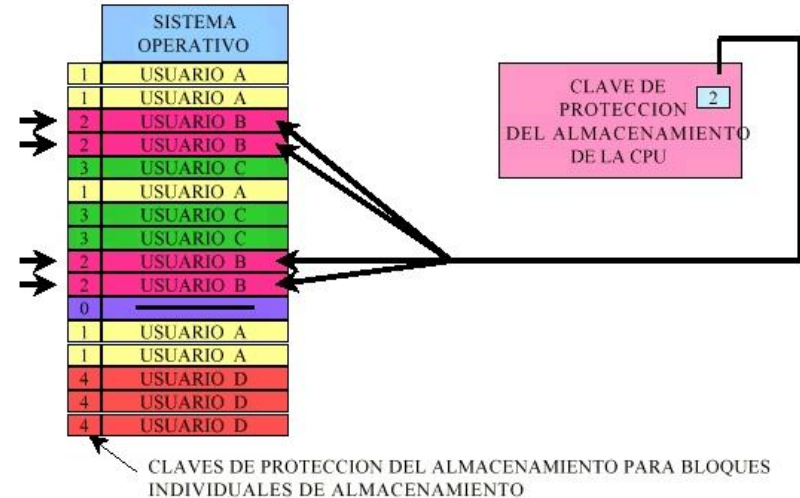
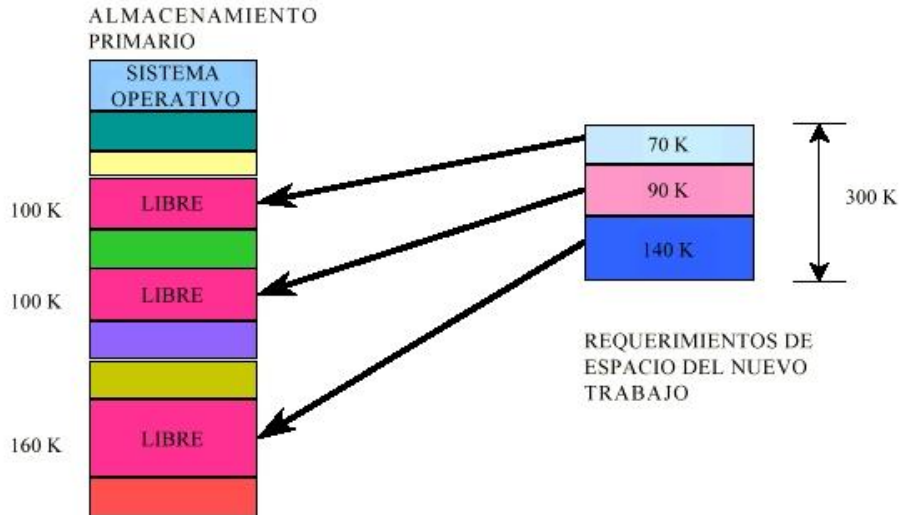


# Segmentación

Los bloques:

- No necesitan ser de igual tamaño.
- Los bloques separados no necesitan ser adyacentes.

Existe protección de almacenamiento: Las claves están bajo el control estricto del S.O. Un programa de usuario, a quien corresponde una cierta clave en la cpu, sólo puede hacer referencia a los otros bloques del almacenamiento con igual clave de protección. Existe fragmentación externa.



# Segmentación

Una dirección virtual es un par ordenado  $v=(s,d)$ :

- “s” es el número del segmento del almacenamiento virtual.
- “d” es el desplazamiento en el segmento.

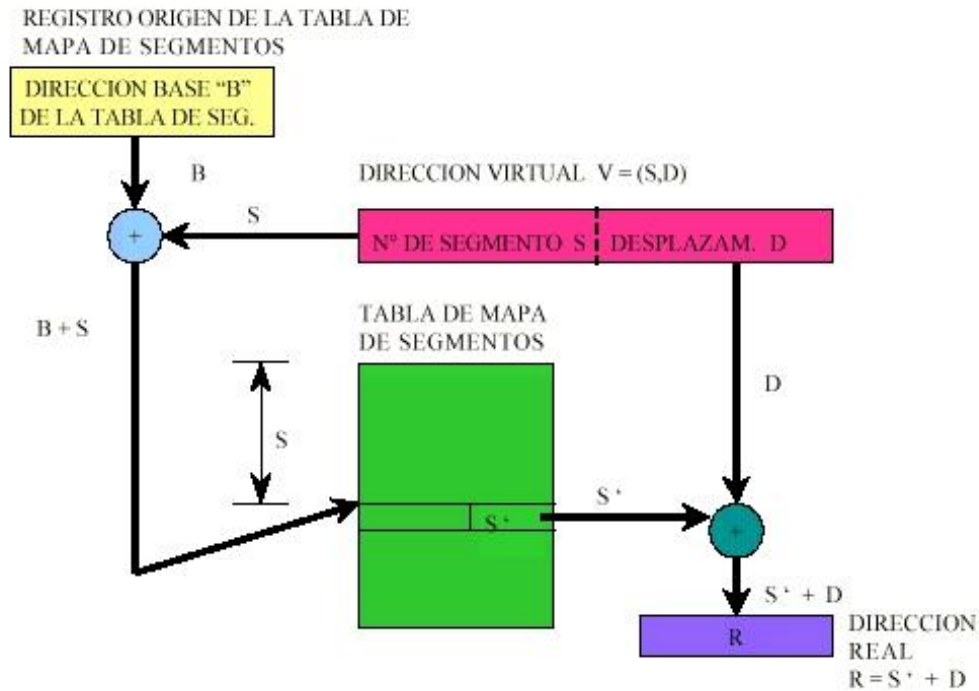
Los segmentos se transfieren del almacenamiento secundario al primario como unidades completas.

Un nuevo segmento puede ser colocado en una serie disponible de posiciones contiguas del almacenamiento primario de tamaño suficiente para alojar al segmento.

**Dirección virtual**

Número de segmento	Desplazamiento
--------------------	----------------

# Traducción de Direcciones de Segmentación por Transformación Directa



# ENTRADA DE LA TABLA DE SEGMENTO

Un “bit de residencia” indica si en la actualidad el segmento se encuentra o no en el almacenamiento primario.

Si el segmento se encuentra en el almacenamiento primario “S ’ ” es la dirección en este almacenamiento donde comienza el segmento.

Si el segmento no se encuentra en el almacenamiento primario “A” es la dirección en el almacenamiento secundario de donde debe recuperarse antes que el proceso pueda continuar.



R = 0 SI EL SEGMENTO NO ESTA EN EL ALMACENAMIENTO PRIMARIO

R = 1 SI EL SEGMENTO ESTA EN EL ALMACENAMIENTO PRIMARIO

BITS DE PROTECCION: (1-SI, 0-NO)

R - ACCESO DE LECTURA

E - ACCESO DE EJECUCION

W - ACCESO DE ESCRITURA

A - ACCESO DE ADICION

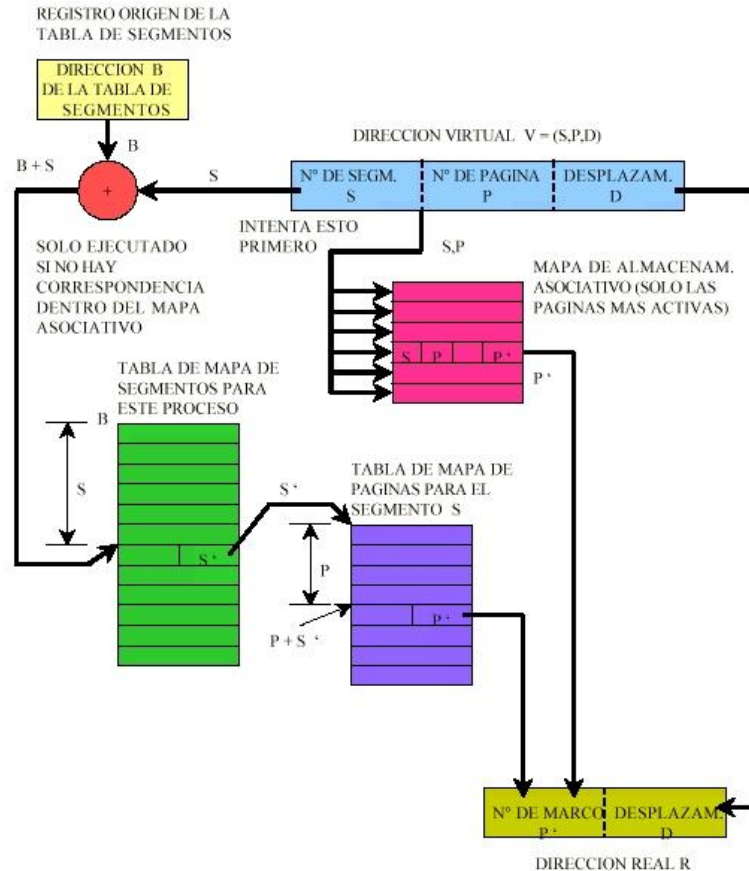
# Sistemas de Paginación / Segmentación

- Ofrecen las ventajas de las dos técnicas de organización del almacenamiento virtual.
- El tamaño de los segmentos es múltiplo del de las páginas.
- No es necesario que todas las páginas de un segmento se encuentren al mismo tiempo en el almacenamiento primario.
- Las páginas de almacenamiento virtual, que son contiguas en este almacenamiento, no necesitan ser contiguas en el almacenamiento real.
- El direccionamiento es tridimensional con una dirección de almacenamiento virtual " $v = (s,p,d)$ ":
  - "s" es el número del segmento.
  - "p" es el número de página.
  - "d" es el desplazamiento en la página donde se encuentra asignado el elemento deseado.



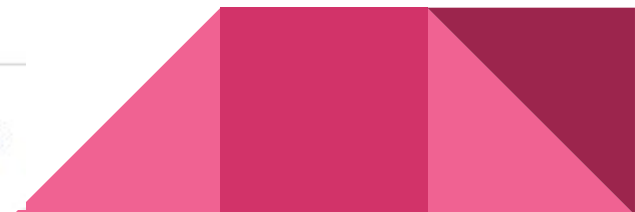


# Traducción de direcciones asociativa directa paginacion y segmentacion



Característica	Segmentación	Paginación	Segmentación Paginada
<b>Unidad de división</b>	Segmentos lógicos de diferente tamaño	Páginas de tamaño fijo	Segmentos que se dividen en páginas de tamaño fijo
<b>Tamaño de las unidades</b>	Variable, según la lógica del programa	Fijo, determinado por el sistema operativo	Los segmentos son variables, pero las páginas son fijas
<b>Gestión de memoria</b>	Se basa en tablas de segmentos	Se basa en tablas de páginas	Usa tablas de segmentos y de páginas
<b>Traducción de direcciones</b>	Dirección lógica: número de segmento y desplazamiento	Dirección lógica: número de página y desplazamiento	Dirección lógica: número de segmento, página y desplazamiento
<b>Fragmentación</b>	Externa (debido al tamaño variable de segmentos)	Interna (si no se usa toda la página)	Menor fragmentación, combinando características de ambas
<b>Ventajas</b>	Organización lógica del programa, fácil protección de segmentos	Gestión eficiente de memoria sin fragmentación externa	Aprovecha las ventajas de ambas técnicas
<b>Desventajas</b>	Fragmentación externa, gestión más compleja	Fragmentación interna, no refleja la estructura lógica del programa	Complejidad mayor en la traducción de direcciones

## TABLA COMPARATIVA



# BENEFICIOS DE LA MEMORIA VIRTUAL

- **Aprovechamiento eficiente de la memoria:** la memoria virtual permite a los sistemas utilizar de manera más eficiente la memoria física. Al dividir la memoria en páginas, el sistema operativo puede cargar solo las partes necesarias de un programa en la memoria, lo que mejora el rendimiento general del sistema.
- **Facilidad en la ejecución de programas grandes:** La memoria virtual facilita la ejecución de programas más grandes que la memoria física disponible, permitiendo que las aplicaciones utilicen más memoria de la que está físicamente presente al utilizar una combinación de RAM y almacenamiento secundario (como discos duros).
- **Aislamiento y protección:** La memoria virtual ofrece beneficios de seguridad y aislamiento entre procesos. Esto evita que los programas accedan a las áreas de memoria de otros procesos, lo que refuerza la protección y estabilidad del sistema.
- **Multiprogramación eficiente:** La memoria virtual facilita la multiprogramación al permitir que varios programas se ejecuten simultáneamente sin interferir entre sí, gracias a la gestión dinámica de la memoria y el uso de paginación o segmentación.



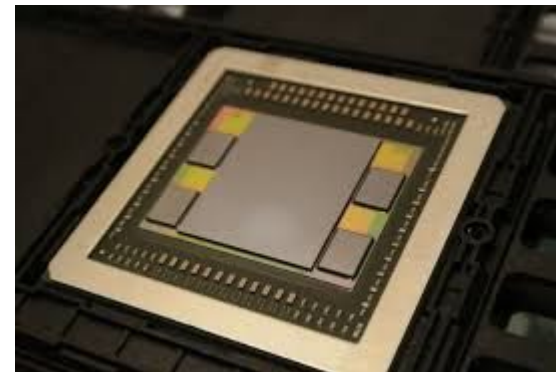
# CONTRAS DE LA MEMORIA VIRTUAL

- **Sobrecarga en la administración de la memoria:** Administrar tablas de páginas y manejar las interrupciones causadas por fallos de página puede consumir recursos considerables del sistema, lo que puede afectar el rendimiento general.
- **Rendimiento más lento:** Esto ocurre cuando el sistema necesita cargar y descargar continuamente páginas de la memoria secundaria, lo que es mucho más lento que acceder a la RAM directamente.
- **Fragmentación:** Esto sucede cuando el tamaño de las páginas no coincide exactamente con el tamaño de los bloques de memoria solicitados por los programas, lo que lleva a un desperdicio de espacio dentro de las páginas asignadas.
- **Requerimiento de hardware especializado:** Requiere hardware especializado, como una unidad de gestión de memoria (MMU, por sus siglas en inglés). Esto aumenta la complejidad y el costo de los sistemas, tanto a nivel de hardware como de diseño de software.
- **Complejidad de diseño e implementación:** Los sistemas deben gestionar correctamente fallos de página, tablas de páginas y sistemas de caché para evitar problemas graves de rendimiento o seguridad.



# Soporte de hardware necesario para la memoria virtual

- **Unidad de Gestión de Memoria (MMU):** Traduce direcciones lógicas a físicas.
- **Tablas de Páginas:** Soporte para la gestión de tablas que mapean direcciones lógicas a físicas.
- **TLB (Translation Lookaside Buffer):** Caché para traducciones rápidas de direcciones.
- **Control de Fallos de Página:** Manejo de interrupciones cuando ocurre un fallo de página.
- **Soporte para Paginación y Segmentación:** Dividir la memoria en páginas o segmentos y gestionarlos de manera eficiente.
- **Almacenamiento Secundario:** Uso de discos duros o SSD para extender la capacidad de la memoria física.



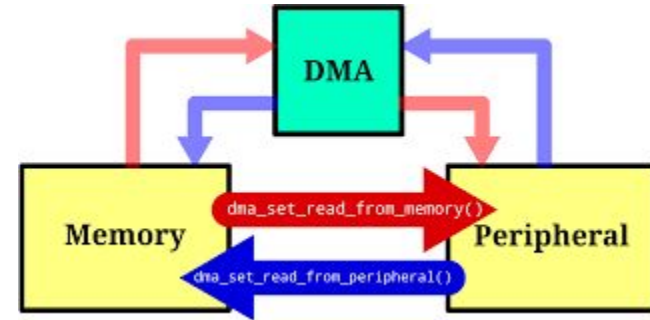
# MEMORIA VIRTUAL EN LINUX - WINDOWS

Aspecto	Linux	Windows
Sistema de memoria virtual	Basado en <b>paginación</b> con soporte para <b>swap</b> .	Basado en <b>paginación</b> con uso del archivo de paginación ( <b>pagefile.sys</b> ).
Archivo/página de intercambio	Utiliza particiones o archivos de <b>swap</b> para ampliar la memoria física.	Utiliza un archivo de paginación ( <b>pagefile.sys</b> ) para gestionar la memoria virtual.

# Acceso directo a Memoria DMA

El acceso directo a memoria permite a cierto tipo de componentes de una computadora acceder a la memoria del sistema para leer o escribir independientemente de la unidad central de procesamiento (CPU). Muchos sistemas hardware utilizan DMA, incluyendo controladores de unidades de disco, tarjetas gráficas y tarjetas de sonido. DMA es una característica esencial en todos los ordenadores modernos, ya que permite a dispositivos de diferentes velocidades comunicarse sin someter a la CPU a una carga masiva de interrupciones.

Una transferencia DMA consiste principalmente en copiar un bloque de memoria de un dispositivo a otro. En lugar de que la CPU inicie la transferencia, la transferencia se lleva a cabo por el controlador DMA.



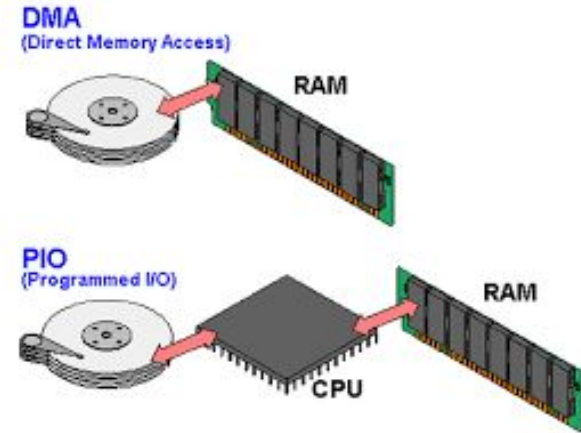
# DMA Secuencia de Eventos

Una operación de E/S por DMA se establece ejecutando una corta rutina de inicialización. Consiste en varias instrucciones de salida para asignar valores iniciales a:

- AR: Dirección de memoria de la región de datos de E/S IOBUF (buffer de entrada/salida).
- WC: Número N de palabras de datos a transferir.

Una vez inicializado, el DMA procede a transferir datos entre IOBUF y el dispositivo de E/S. Se realiza una transferencia cuando el dispositivo de E/S solicite una operación de DMA a través de la línea de petición del DMAC.} Después de cada transferencia, se decrementa el valor de WC y se incrementa el de AR.

La operación termina cuando  $WC=0$ , entonces el DMAC (o el periférico) indica la conclusión de la operación enviando al procesador una petición de interrupción.

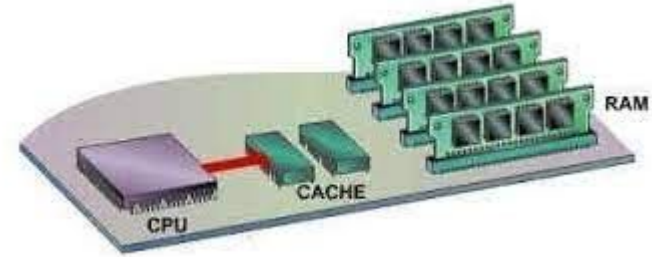




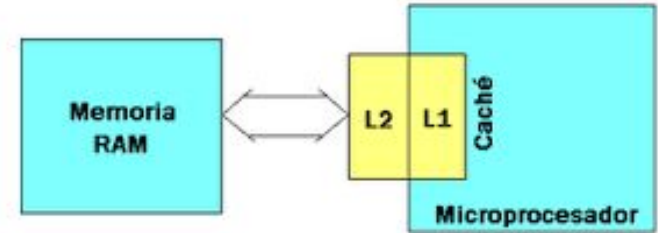
# Memoria Caché

La memoria caché es la memoria más rápida del sistema de memoria.

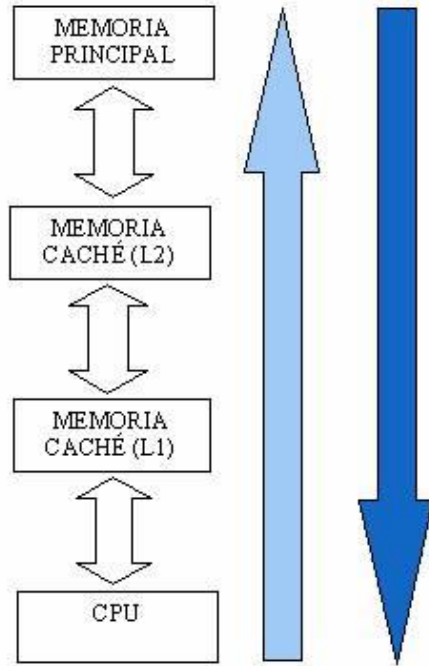
Cada vez que la CPU intenta acceder a una dirección de memoria que no está en ningún bloque de la caché se produce un fallo de caché, y en caso contrario un acierto de caché. La consecuencia del fallo es que se suspende momentáneamente el acceso, se copia el bloque que contiene la dirección desde la memoria principal a la memoria caché y a continuación se reanuda el acceso.



**Memoria Caché**



# Organización de caché - Niveles de Caché



Inicialmente cuando se introdujeron las cachés, el sistema típico tenía una única caché. Sin embargo recientemente lo habitual es disponer de dos niveles de caché o incluso tres en computadores de gama alta. Estos niveles se denominan L1,L2,etc., donde el nivel L1 es el más cercano al procesador.

El objetivo de la caché L2 es reducir la penalización debida a los fallos de caché del nivel L1. En teoría, cuantos más niveles de caché haya entre el procesador y la memoria principal mejor rendimiento proporciona el sistema de memoria pero con un coste mayor.

# Organización de caché

## Caché unificada

La caché unificada sirve tanto para datos como para código. Tiene la ventaja de que se reparte automáticamente los bloques cacheados entre datos y código. De tal forma que si son necesarios menos bloques de datos y más de código estos se reparten de forma eficiente y transparente.

## Caché dividida

La memoria caché dividida está formada por dos cachés, una para datos y otra para código. Podría suceder que la memoria caché de datos tuviese bloques libres o muy poco usados, mientras que la memoria caché de código estuviese generando fallos de caché. En este caso el rendimiento sería menor.



# Estrategia de Reemplazo

Las estrategias de reemplazo establecen qué bloque de la memoria caché debe reemplazarse por un bloque de memoria principal cuando se producen un fallo de caché., Esto sólo tiene sentido cuando el bloque de memoria principal puede ir a más de un bloque de la memoria caché.

- **LRU (Least Recently Used):** El bloque a reemplazar es el que lleva más tiempo sin haber sido accedido. Por tanto, una caché que use esta estrategia, debe mantener esta información de alguna forma.
- **Reemplazo aleatorio:** En el caso del reemplazo aleatorio se elige de forma aleatoria entre uno cualquiera de los bloques de caché candidatos.



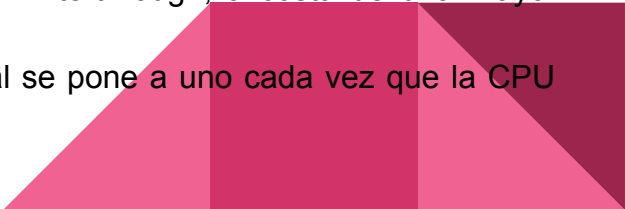
# Estrategia de Escritura

Establecen la relación entre las escrituras en la memoria caché y la memoria principal. El objetivo fundamental de estas estrategias es garantizar la coherencia entre la información de la caché y la memoria principal, afectando lo mínimo posible al rendimiento del sistema.

- **Write-through o escritura a través:** La estrategia de escritura write-through también se denomina escritura a través. En ella, cada vez que se escriba un dato en memoria caché, se modificará inmediatamente en memoria principal. Esto elimina problemas de coherencia, es decir, que cuando un periférico acceda a memoria, siempre accederá a datos actualizados, con lo cual se evita el tener que comprobar si el dato está cacheado y si es diferente al que hay en la memoria principal.
- **Write-back o escritura diferida:** Con la escritura diferida, los datos inicialmente sólo se escriben en la caché. El dato escrito aparece reflejado en la memoria principal sólo cuando el bloque que lo contiene es reemplazado. Con la escritura diferida aparecen problemas de coherencia.

Comparativamente, la estrategia write-back proporciona un mayor rendimiento que la write-through, a costa de una mayor complejidad hardware para el mantenimiento de la coherencia.

Conviene resaltar que se asocia un bit más a cada bloque de caché, el bit dirty, el cual se pone a uno cada vez que la CPU escribe en el bloque de caché.




# PRACTICA CON LINUX Y WINDOWS

**Linux:** Uso de Comandos top, htop, vmstat y free.

**Windows:** Monitor de recursos mediante el administrador de tareas.

Ajustes de Memoria Virtual en Windows.

En Configuración avanzada del sistema, Pestaña Opciones avanzadas, Rendimiento, Configuración, Opciones avanzadas, Memoria virtual. Observa cómo está configurada la memoria virtual.



## BIBLIOGRAFÍA

- Tanenbaum, A. S. (2003). *Sistemas operativos modernos*. Pearson Educación.
- Stallings, W. (2004). *Sistemas operativos* (Vol. 2). Prentice Hall.
- Silberschatz, A., Galvin, P. B., & Gagne, G. (2006). *Fundamentos de sistemas operativos*. McGraw Hill.

